

B.D.U.C.

BETA DISK USERS CLUB

BETA DISK NEWSLETTER NO. 3

Welcome to this the third BDOC newsletter. We also welcome new members in Europe. BDOC now has members in France, Germany, Belgium and Denmark. Thanks to members who have sent contributions, I should be able to feature most of them in forthcoming issues. Apologies for the late return of your contribution disks and tapes.

All members are invited to contribute, I can accept text files in the following formats, Tasword 2 and 3, The Writer and The Last Word. Both 3.5" and 5.25" disk formats may be submitted. Any Beta interface related material will be reviewed for inclusion.

BDOC has no information about TR's business since the last newsletter, but Cumana Ltd. (who it seems manufactured the floppy controller PCB for Technology Research) are repairing Beta+ and Beta 128 interfaces. Service may be obtained by sending your interface with a description of the fault to Cumana at the following address: Cumana Ltd.,

Service Department,
Pines Trading Estate,
Guildford,
Surrey.
GU3 3BH.

There is a minimum charge of £15+VAT. The minimum charge is then plus parts plus return carriage. You should send your interface by recorded post or secured delivery. Cumana Ltd. may be contacted on 0483-503121.

European users should contact Cumana Ltd. for carriage and terms of payment.

You are advised that before sending your unit for repair the cost of repair may well exceed the cost of purchasing a second hand unit. The FDC chip costs about £15 or £20 and the TRDOS EPROM costs £12, Beta interfaces are usually available for £35 approx.

There is a small correction to Nick Cooper's Autoboot program from the second issue, line 225 should be changed to:

```
225 IF z$=CHR$ 8 AND n=1 THEN LET n=INT (f!/16)+1: POKE x+4,n-  
1: RANDOMIZE USR x: LET F=1: GO TO 60
```

THE SECRETS OF TRACK 0. BY HENDRICK BROOTHAERS.

What follows relates to a "dual density" interface.

First some basic things:

- The use of track zero is the same for each type of format, 40 or 80 tracks and single or double sided. This is necessary since first we have to read what kind of format is used before we can use this information for other purposes.
- Each track is divided in 18 sectors.
- Each sector contains 256 bytes.

Use of track zero:

- Only nine of the sixteen sectors are used.
- Sector eight contains all information about the disk such such as name, password (earlier versions only), number of files etc.
- Sectors zero to seven contain information about the files on the disk.
- Each file uses sixteen bytes, thus we can store information for sixteen files in each sector.
- Because we use eight sectors (0 to 7) we can store information for a maximum of 128 files on track zero.

Sector eight contains general information.

- Only the last 31 bytes have a meaning. The first 225 bytes are all zero.

BYTE MEANING

| | | |
|-----|-------------------------|---------------------------------|
| 225 | sector | } First free sector and |
| 226 | track | } First free track on the disk. |
| 227 | format type | (see following table) |
| 228 | number of files | (deleted files included) |
| 229 | } | number of free sectors |
| 230 | } | on the disk. |
| 231 | 16 | |
| 232 | 0 | |
| 233 | 0 | |
| 234 | to 242 | password (8 chars long) |
| 243 | 0 | |
| 244 | number of deleted files | |
| 245 | to 253 | name of the disk (8 chars long) |
| 254 | 0 | |
| 255 | 0 | |

Table for byte 227:

22 = 80 track/double side = 2560 sectors
23 = 40 track/double side = 1280 sectors
24 = 80 track/single side = 1280 sectors
25 = 40 track/single side = 640 sectors

Sectors zero to seven contain file information.

- As mentioned before, each file uses 16 bytes.
- Some of these bytes have a different meaning, depending on the type of file. (Basic, Code or Data)

BYTE MEANING

0 to 7 = file name
8 = file type B(asic) C(ode) D(ata)
9 to 10 = load address for a Code file
= total length for a Basic file
= address for a Data file
11 to 12 = run address for a Code file
= program length for a Basic file
= array length for a Data file
13 = length in number of sectors
14 sector }
15 track } where the file is located on the diskette

REMARKS:

- For a DELETED file byte 1 is set to value 1.
- The RUN line number for a BASIC program is directly behind the actual basic program on the disk, and NOT on track zero.
- The end of the directory is recognized when byte 1 of a program name has the value zero.

The next part describes using machine code calls and Beta DOS it was written as a reference for earlier versions of DOS. Some features of these early versions are included in 5.03 but the call address may differ.

MACHINE CODE AND BETA DOS.

| FUNCTION: | 3.0 | 4.03 | 4.06 | 4.07 | 4.10 | 4.1 | 4.11 | TRDOS VERSION |
|-------------|-------|-------|-------|-------|-------|-------|-------|------------------|
| | | 4.04 | | 4.09 | | | 4.12 | |
| DOS ROM ON | 3CAAH | 15487 | 15487 | 15467 | 15467 | 15467 | 15467 | 15467 |
| READ DISK | 3C09H | 11958 | 11992 | 11958 | 11991 | 11934 | 11990 | |
| WRITE DISK | 3C0CH | 11974 | 12008 | 11974 | 12007 | 11950 | 12027 | |
| DOS ROM OFF | 3CBEH | 15484 | 15484 | 15484 | 15484 | 15484 | 15484 | |

To use the read and write calls set the registers as follows:

DE = D =track, E =sector
 HL = buffer address
 BC = B number of sectors to do, C=0

Don't forget to turn TRDOS on/off before/after using.

The following group may be used to address the FDC (Floppy Disk Controller Chip) directly.

PORT OUT

PORT IN

| | |
|----------------------------|----------------------------|
| 1F COMMAND to command reg. | 1F STATUS from status reg. |
| 3F TRACK to track reg. | 3F TRACK from track reg. |
| 5F SECTOR to sector reg. | 5F SECTOR from sector reg. |
| 7F DATA to data reg. | 7F DATA from data reg. |

The following controls the BETA interface port, FC OUT generates a clock to enable some circuits in the BETA interface.

F7 }

)-These two during OUT generate a clock to the interface to
 FF } stores info such as drive number/side/master reset to FDC.

-during IN they check if a DRQ (data request) or an INTRQ
 (interrupt request) is pending from the interface. FF is
 also used to switch the BETA ROM on and off.

The following BETA DOS system variables are for the earlier versions mentioned above.

BETA DOS VARIABLES.

5CC8 Format for drive A
 5CC9 Format for drive B
 5CCA Format for drive C
 5CCB Format for drive D

BETA DOS VARIABLES continued.

5CCC Track to read from
5CCD Sector to read from
5CCE 00=read / FF=write

5CD9/A Effective load address
5CDB/C Effective length
5CCD File name (8 chars long)
5CE5 File type B/C/D/*
(B:Basic C:Code D:Array #:Print file)

5CE6/7 Load address for code file/total length for basic
5CE8/9 Program length for basic or length for code file
5CEA File length in sectors
5CEB/C Sector and track start of file

5CF6 Drive number
5CF8 Drive to read from
5CF9 Drive to write to

5D02 Temporary storage for address for text to be printed

5D19 Drive number Drive A=0, B=1, C=2, D=3

BETA DISK INTERFACE HARDWARE - PART 1 LOGIC AND CONTROL.

The circuit diagram in this issue relates to BETA 128 with 5.xx hardware. It is known that many revisions to the hardware were carried out and this diagram may not be accurate, but it does provide a useful illustration of BETA hardware principles.

Component List.

1 27128 EPROM 16384 WORDS X 8 BITS NEC D27128D
3 74LS32 QUAD 2-INPUT OR
1 74LS74 FLIP-FLOP D-TYPE
1 74LS30 8-INPUT NAND
1 74LS04 BUFFER HEX INVERTING
1 74LS123 DUAL MONOSTABLE MULTIVIBRATOR

THE LAST WORD BETA EXTENSION. BY M.J.SMITH BDOC.
BETA 128 5.xx DOS.

The Last Word Beta extension as supplied is coded for version 4.xx DOS. To enable correct operation of the CAT command from TLW the overlay code must be loaded as described and the following patch included. Also the Beta lines at 4000 must be inserted EXACTLY as shown below. This is required since TLW code inserts the text file title to be loaded or saved directly into the BASIC lines for execution it also preserves the return address and executes a RST 8 with the PRINT USR 8 command in line 4020. The sequence to modify your TLW is as follows:

With your configured TLW resident return to BASIC. Load the Beta extension code from tape. LOAD "BETA2CODE" CODE this loads 488 bytes at address 50000, now in direct mode POKE 50112,61. Add or edit the existing lines as below.

```
40 CLEAR 26500: GO SUB 100: RANDOMIZE USR 15619: REM: LOAD
"TLW2"CODE
45 GO TO 1000
68 RANDOMIZE USR 15619: REM: ERASE "boot"
69 RANDOMIZE USR 15619: REM: ERASE "TLW2"CODE
70 RANDOMIZE USR 15619: REM: SAVE "boot" LINE 40
80 RANDOMIZE USR 15619: REM: SAVE "TLW2"CODE 50000,15535
4000 REM BETA LINES
4010 RANDOMIZE USR 15619: REM: SAVE "A:12345678"CODE 12345,
12345
4020 PRINT USR 8
```

The values in quotes and the code start and length in line 4010 are dummy, TLW overwrites these with the current filename to load or save.

The configured code and the boot loader can be saved or resaved by GO TO 68 (resave) or GO TO 70 (save).

NOTE: Resave deletes the original versions on the disk!

The overlay code contains pointers to the Beta DOS call routines starting at 50440 is LOAD, 50442 is ERASE, 50444 is SAVE, 50446 is CAT.

The Beta prompt text strings start at 50450.

BDOC, 2, DOWNHAM AVENUE, RAWTENSTALL, ROSSENDALE, LANCASHIRE. BB4 8JY